Measurement-based quantum computation on graph states and undecidable logic theories

Maarten Van den Nest[1] and Hans J. Briegel[1,2]
[1] *Institut für Quantenoptik und Quanteninformation der Österreichischen Akademie der Wissenschaften, Innsbruck, Austria*
[2] *Institut für Theoretische Physik, Universität Innsbruck, Technikerstraße 25, A-6020 Innsbruck, Austria*
(Dated: October 9, 2006)

We establish a connection between measurement-based computation on graph states and the field of mathematical logic. We show that the computational power of graph states as resources for measurement-based quantum computation is reflected in the expressive power of (classical) formal logic languages defined on the underlying graphs. In particular, it is shown that for all graph states which disallow efficient classical simulation of measurement-based quantum computations, the underlying graphs are associated with undecidable logic theories. Here undecidability is to be interpreted in the sense of Gödel, meaning that there exist propositions, expressible in the above classical formal logic, which cannot be proven or disproven.

Introduction.— Quantum computers are devices that use quantum mechanics for enhanced ways of information processing. While a number of examples can be given where quantum computers outperform classical devices, it remains a central problem to pin down the essential features that give quantum computers their additional power [1, 2, 3, 4].

There are strong indications that, in this context, quantum entanglement is a key resource. The role of entanglement in quantum computation is made particularly explicit in the one-way model of a quantum computer, introduced in Ref. [5]. In this model, the entire resource of a quantum computation is given in form of a specific entangled state, the 2D cluster state [6], of a large number of qubits. The cluster state is known to be a universal resource for measurement-based quantum computation (MQC) [5, 7]. Quantum computation on cluster states proceeds by measuring the qubits of the cluster, one by one, in a specific order and basis, and by classical processing of the measurement results.

As entanglement can only decrease in a one-way computation, the enhanced computational power of such a quantum computer (beyond a classical Turing machine) must originate in the entanglement structure of its resource state. Owing to this insight, a series of papers (see Ref. [8] and Refs. within) have recently been devoted to analyzing the entanglement of cluster and similar multi-partite entangled states, called *graph states* [8], and the use of graph theoretical concepts in this context has been quite fruitful. A graph state on n qubits is defined by means of a graph on n vertices, which completely encodes the correlations in the system. One has found several entanglement measures which grow unboundedly on the 2D cluster states and certain graph states, while they remain constant on other sets of states. Important examples are the *entanglement width* measures [3, 4], which have graph theoretical roots [9]. In previous work we showed that the entanglement width is unbounded not only on the family of 2D cluster states, but must be unbounded on any family of resource states which is a universal resource and/or which disallows efficient classical simulation of MQC [3, 4].

The above results provide criteria, in terms of entanglement, to assess the computational power of resource states, hence contributing to the central issue of understanding which features give quantum computers their additional powers over classical devices. In this letter we continue the study of this issue, focusing on graph state resources, as these are natural generalizations of the cluster states. In the following we will establish new necessary conditions for graph state resources to yield a computational speed-up with respect to classical computers, which are entirely different in nature with respect to previously established requirements. In particular, we will show that the above entanglement width criteria allow one to establish a connection between MQC on graph states and the properties of the associated graphs in relation with mathematical logic theory—more particularly, *decidability* of logic theories.

As is well known since Gödel's incompleteness theorems [10], every formal system that is sufficiently interesting (or rich) contains statements which can neither be proved to be true nor to be false within the axiomatic framework of the system. Gödel's incompleteness theorem not only applies to formal systems relating to natural numbers (cf. Peano arithmetic) but also to graphs. Indeed, many interesting graph properties can be expressed within a formal language, denoted by $\mathcal{L}$ (the exact definition of this language is stated below). Examples of such properties are planarity or 2-colorability of graphs. In this letter we will show that the computational power of graph states as resources for measurement-based quantum computation is reflected in the expressive power of such formal language $\mathcal{L}$ defined on the underlying graphs. In particular, the following results are obtained.

Theorem 1. *For all graph state resources that are universal for measurement based quantum computation, the logic $\mathcal{L}$ defined on the underlying graphs must be undecidable.*

Theorem 2. *For all graph state resources which disallow efficient classical simulation of measurement based quantum computation, the logic $\mathcal{L}$ defined on the underlying graphs must be undecidable.*

Here undecidability is to be interpreted in a sense similar to Gödel, meaning that there exist propositions, expressible in the logic $\mathcal{L}$, which cannot be proven or disproven. Universal-

ity of resources is defined as in Ref. [3], see also below.

Theorems 1 and 2 provide necessary conditions to assess the computational power of graph state resources by considering the underlying graphs, which, by the very definition of graph states, render a classical encoding of the quantum correlations in the states. The present results state that *graph states can only yield a computational speed-up with respect to classical computers, if the underlying graphs are such that the (classical) formal language $\mathcal{L}$ defined on them is undecidable*. Such undecidability is to be regarded as a notion of complexity of the graphs—and hence of the correlations in the system—stated independently of any quantitative measure such as the entanglement width measures.

This paper aims at connecting two quite different fields of research, namely quantum computation and mathematical logic. Therefore, in the following we will give a brief review of the basic concepts of measurement-based (one-way) quantum computation, as well as logic theories on graphs. This will allow us to reformulate the main theorems in a precise manner. The proofs of theorems 1 and 2 are obtained by combining our previous results regarding entanglement width and MQC on graph states [3, 4], and a recent graph theoretic result [11]. We will conclude the paper with an interpretation of these results.

Graph states and MQC.— We will be concerned with a particular class of multi-party quantum states, called *graph states*, which are generalizations of the 2D cluster states. A graph state $|G\rangle$ on m qubits is the joint eigenstate with eigenvalue one of m commuting correlation operators

$$K_a := \sigma_x^{(a)} \bigotimes_{b\in N(a)} \sigma_z^{(b)}, \tag{1}$$

where σ_x and σ_z are Pauli matrices, and the upper indices denote on which qubit system these operators act. Moreover, $N(a)$ denotes the set of neighbors of qubit a in the graph G [8]. Thus, a system in a graph state has $\langle K_a\rangle = 1$ for every a.

For example, a 2D cluster state is obtained if the underlying graph is a $k \times k$ square lattice $C_{k\times k}$ (thus $m = k^2$).

As graph states are generally highly entangled, and as they can efficiently be prepared by applying a suitable poly-sized quantum circuit to a product input state [8], they form natural candidates to serve as resources for MQC. We envisage a situation where an (infinitely large) family of graph states $\Psi = \{|G_1\rangle, |G_2\rangle, \dots\}$ with growing system size is considered, and where local measurements can be implemented on arbitrary members of Ψ, thus implementing quantum computations. We call Ψ a universal resource for MQC if every possible quantum state can be prepared deterministically by performing local operations and classical communication (LOCC) on members of Ψ [3]. Note that a weaker definition of universality can be given, as in Ref. [2]; this, however, does not affect the results in this paper. Universality is a property which is attributed to an infinite family of states Ψ, and not to a single quantum state, as arbitrarily large computations need to be accounted for. The canonical example of a universal resource is the family of $k \times k$ cluster states $\{|C_{k\times k}\rangle\}_k$ [5].

Further, we will say that efficient classical simulation of MQC on a family of states Ψ is possible, if for every state $|G_i\rangle \in \Psi$ it is possible to simulate every LOCC protocol on a classical computer with overhead poly(m_i), where m_i denotes the number of qubits on which the state $|G_i\rangle$ is defined [4]. Evidently, resources for which efficient simulation of MQC is possible, cannot offer any computational speed-up over classical computers. Note also that, while universality and classical simulatability are closely related issues, they are principally distinct; we refer to Ref. [4] for an extensive discussion.

In the following we will focus on the graphs underlying families of graph states, and the formal languages defined on them. Note that by definition (1) the graph G is an encoding of the correlations present in the corresponding graph state. Therefore, any property of these graphs reflects a property of the corresponding states, and, more particularly, the correlations in these states.

Graphs and logic.— Next we define some basic notions of logic theory which are necessary to state our main results concisely below. We refer to Refs. [9, 12] for an extensive treatment. We also emphasize that we will favor clarity of the exposition over mathematical rigor.

First we define *relational structures*. Let D be a finite or countable set. A function $A : D^m \to \{\text{true}, \text{false}\}$, where D^m is the set of all m-tuples of elements of D, is called a *relation symbol with arity* m. Thus, A relates tuples of m elements in D with each other. A pair $S = \langle D, A\rangle$ is then called a *relational structure* on the domain D if D is a finite or countable set and the A is a relation symbol on D [13]. As a simple example of a relational structure, define the relation symbol `succ` with arity 2 on the domain of natural numbers $\mathbb{N}$ by

$$\texttt{succ}(x, y) = \begin{cases} \text{true} & \text{if } y = x + 1 \\ \text{false} & \text{otherwise,} \end{cases} \tag{2}$$

for every $x, y \in \mathbb{N}$. Then $\langle \mathbb{N},\ \texttt{succ}\rangle$ is a relational structure. As a second, and for our purposes more relevant example, we see how graphs can be naturally be cast as relational structures. Let $G = (V, E)$ be a graph with vertex set V and edge set E. First, define a relation symbol `edge` of arity 2 on V by

$$\texttt{edge}(a, b) = \begin{cases} \text{true} & \text{if } \{a, b\} \in E \\ \text{false} & \text{otherwise,} \end{cases} \tag{3}$$

for every pair of vertices $a, b \in V$. Then $\langle V,\ \texttt{edge}\rangle$ is a relational structure called an *adjacency structure*. Note that adjacency structures entirely encode the underlying graphs.

One of the main reasons to consider graphs (or other objects) as relational structures, is that this approach allows one to formalize properties such as 2-colorability, connectedness, etc. This formalization is obtained by defining a logical calculus in which such graph properties can be expressed. Roughly speaking, a logic on a relational structure S corresponds to a

set of rules which determine the basic constituents with which statements regarding S can be constructed. Such formalization in terms of logic allows, in principle, artificial devices to mechanically prove or disprove, for a given graph or set of graphs, properties expressible in this logic.

The simplest logic is *first-order logic*, which is defined as follows. Let $S = \langle D, A\rangle$ be a relational structure. A variable x is called a first-order variable if it denotes an element of D. First-order logic formulas on S are logic formulas written by using the relation symbol A, first-order variables, quantifiers $\forall$ and $\exists$ over first-order variables, and connectives $\neg$, $\vee$, and $\wedge$. A simple example of a first-order logic formula on the structure $\langle \mathbb{N},\ \texttt{succ}\rangle$ is the formula

$$\forall x\ \exists y\ \exists z\ \texttt{succ}(x, y) \wedge \texttt{succ}(y, z), \tag{4}$$

which expresses that every natural number x has a successor $y = x + 1$ and a second successor $z = x + 2$.

Regarding graphs as relational structures, it turns out that first-order logic is often not rich enough to express interesting graph properties. One therefore extends first-order logic by allowing, next to first-order variables, also *set variables* X, Y, Z (indicated by capital letters), which denote subsets of D; furthermore, one adds quantifications $\forall X$, $\exists X$ over set variables, and also atomic formulas of the form $x \in X$, where x is a first–order variable and X is a set variable. The logical calculus which is thus obtained is called *monadic second-order logic*, or *MS logic* in short. MS logic is strictly more expressive than first-order logic, i.e., there are problems which can be expressed with MS logic which cannot be expressed using only first-order logic. An example of an MS formula on an adjacency structure $\langle V,\ \texttt{edge}\rangle$ is

$$\begin{aligned}\exists X\ \exists Y\ \{&\forall z\ (z \in X \vee z \in Y)\ \wedge \\ &\forall z\ \forall z'\ \neg\texttt{edge}(z, z') \vee \neg(z, z' \in X \vee z, z' \in Y)\},\end{aligned}$$

which expresses that the graph underlying $\langle V,\ \texttt{edge}\rangle$ can properly be colored with 2 colors (i.e., the vertices can partitioned in two classes such that no two adjacent vertices are in the same class). It turns out that many interesting graph properties can be expressed in MS logic, among which there are several NP-hard problems (such as e.g. 3-colorability), indicating that MS logic on graphs has a considerable expressive power.

A slight extension of MS logic is obtained by including atomic formulas of the form $\texttt{Even}(X)$, indicating that the set X has even cardinality. In this way one obtains MS logic with the additional possibility to count modulo two, denoted by *C_2MS logic* [11], which will be our topic of interest (i.e., it corresponds to the logic $\mathcal{L}$ as denoted above). C_2MS logic e.g. allows one to express whether two graph states are local unitary equivalent [14].

Decidability of logic theories.— Next we introduce the fundamental issue of *decidability* of logic theories. We again refer to [9, 12] for details. Let $\mathcal{S} = \{S_1, S_2, \dots\}$ be a (finite or infinite) family of relational structures $S_1, S_2, \dots$, and let $\mathcal{L}$ be a logic defined on $\mathcal{S}$. We will be interested in the case where $\mathcal{S}$ is a set of graphs (adjacency structures) and $\mathcal{L}$ is C_2MS logic. The set $\mathcal{S}$ is said to have a *decidable* $\mathcal{L}$-theory if for every formula ϕ expressed in the logic $\mathcal{L}$, it is possible to decide (in finite time) whether there exists a relational structure $S_\alpha \in \mathcal{S}$ for which ϕ is satisfied [15]. The set $\mathcal{S}$ is said to have an *undecidable* $\mathcal{L}$-theory if it does not have a decidable $\mathcal{L}$-theory.

For example, in the case of graphs the formula ϕ could correspond to graph planarity, 2-colorability, etc. Then the question is asked whether there exists a graph in a given family of graphs which is planar, 2-colorable, etc. If every possible question, that is, every formula expressible in the language $\mathcal{L}$, can be answered in finite time, then this family is said to have a decidable $\mathcal{L}$-theory. Note that the decidability or undecidability of a logic theory $\mathcal{L}$ on a set $\mathcal{S}$ is a reflection of both the expressive power of the logic $\mathcal{L}$ and the complexity (regarded in a colloquial sense) of $\mathcal{S}$.

Before giving examples of (un)decidable MS theories, is important to make the following two remarks. First, decidability of a logic theory is not concerned with the *efficiency* with which problems can be solved—one only asks whether it is *in principle* possible to verify whether a given formula ϕ is true, where one does not care about e.g. the computational complexity of a possible verification algorithm. Second, note that *any* first-order or MS theory is decidable on structures $\mathcal{S} = \{S_1, S_2, \dots, S_N\}$ where *both the number of relational structures N is finite and every S_α has a finite domain*. This is simply because, in this finite regime, any formula can be verified by an exhaustive enumeration of cases. Thus, decidability is only relevant when infinite structures are considered (i.e., either $|\mathcal{S}| = \infty$ or at least one of the $S_\alpha \in \mathcal{S}$ has an infinitely large domain).

Let us now give some important examples. First, let $\mathcal{S}_{\mathbb{N}} := \{\langle \mathbb{N},\ \texttt{succ}\rangle\}$ consist of a single relational structure, where the relation symbol succ was defined above; note that the domain $\mathbb{N}$ has infinite cardinality. It was shown by Büchi that $\mathcal{S}_{\mathbb{N}}$ has a decidable MS theory [16]. Second, let $\mathcal{S}_{\text{bin}}$ be the set of all binary tree graphs, which are regarded as so-called incidence structures. A milestone result was obtained by Rabin, who proved that $\mathcal{S}_{\text{bin}}$ has a decidable MS theory [17]. This result has many important implications in graph theory and computer science. Further, let $\mathcal{S}_{\text{2D}}$ be the set of all 2D cluster graphs. Then $\mathcal{S}_{\text{2D}}$ has an *undecidable* C_2MS theory [18]. As final examples, let $\mathcal{S}_{\text{tri}}$ and $\mathcal{S}_{\text{hex}}$ be the sets of all triangular lattice graphs and hexagonal lattice graphs, respectively, regarded as adjacency structures. Then also $\mathcal{S}_{\text{tri}}$ and $\mathcal{S}_{\text{hex}}$ have *undecidable* C_2MS theories [19].

Main results.— Keeping in mind that the graph states corresponding to the 2D rectangular, hexagonal and triangular lattices have been shown to be universal resources for MQC [3], the above examples already suggest a connection between the computational power of a family of graph states as a resource for MQC, and the C_2MS logic defined on the underlying graphs. This connection will now be fully estab-

lished, as we are now in a position to precisely state and prove the main results of this letter, i.e., theorems 1 and 2. Let $\mathcal{G} = \{G_1, G_2, \dots\}$ be an (infinitely large) family of graphs and let $\Psi(\mathcal{G})$ be the associated family of graph states. Theorems 1 and 2 can then precisely be formulated as follows.

Theorem 1. *If $\mathcal{G}$ has a decidable C_2MS logic theory then $\Psi(\mathcal{G})$ cannot be a universal resource for MQC.*

Theorem 2. *If $\mathcal{G}$ has a decidable C_2MS logic theory then MQC performed on $\Psi(\mathcal{G})$ can classically be simulated efficiently.*

The proofs of theorems 1 and 2 are in fact quickly obtained by invoking previous results of the present authors and a highly nontrivial result from graph theory. In Ref. [11] it was proved that every class $\mathcal{G}$ which exhibits a divergence with respect to a graph invariant called *rank width* (see Ref. [9] for definitions) must have an undecidable C_2MS theory. In previous work [3], the present and other authors proved that any universal resource $\Psi(\mathcal{G})$ must have an unbounded rank width. The proof of theorem 1 is then immediately obtained. Similarly, to prove theorem 2, we use a previous result of ours, stating that every family $\Psi(\mathcal{G})$ with a bounded rank width allows an efficient simulation of MQC [4].

Discussion.— In theorems 1 and 2 the desired connection between measurement based quantum computation on graph states and mathematical logic theories on the underlying graphs is fully established. It is the authors' opinion that the present results should be regarded as conceptual results, aimed at establishing a connection between seemingly remote areas of research, rather than yielding direct practical applications. We are aware that assessing whether a family of graphs has a decidable C_2MS theory is a formidable task, and that logic theory itself is a dynamic area of research with difficult outstanding problems [20]. Therefore the present results are not likely to e.g. directly provide new examples of states on which MQC can be simulated efficiently. Nevertheless, we believe that our findings present a new, and possibly deep, perspective towards understanding the central issue *Which features give quantum computers their additional powers over classical devices?* The present connection to logic theory offers an entirely new view on "how complex" states need to be in order for them to provide computational speed-ups, next to more standard considerations regarding entanglement. While theorems 1 and 2 in fact follow from previously obtained results regarding MQC and entanglement width of graph states [3, 4], the resulting logic criteria are entirely different in nature.

Finally, one might be inclined to relate (C_2MS) logic formulas defined on graphs to the content of the quantum computations (i.e., measurement patterns) implemented on the corresponding graph states. As far as the authors are aware, there does not seem to be a direct relation between the classical logic defined on graphs and the quantum measurements—that might be associated to a quantum logic—on the corresponding graph states. Thusfar it seems that the classical logic theories, and the issue of their (un)decidability, are related to assessing the complexity of the graphs, and hence of the (correlations in the) graph states, with no direct correspondence to quantum algorithms. However, it would be very interesting to investigate this issue in more detail, and we leave this as an open problem.

We thank Kris Luttmer and Sang-Il Oum for interesting discussions. This work was supported by the FWF and the European Union (QICS, OLAQUI, SCALA).

[1] R. Jozsa and N. Linden, Proc. R. Soc. London, Ser. A **459**, 2011 (2003); G. Vidal, Phys. Rev. Lett. **91**, 147902 (2003); I. Markov and Y. Shi, quant-ph/0511069; Y. Shi, L. Duan and G. Vidal, quant-ph/0511070; R. Jozsa, quant-ph/0603163; N. Yoran and A.J.Short, Phys. Rev. Lett. 96, 170503 (2006).
[2] D. Gross and J. Eisert, quant-ph/0609149.
[3] M. Van den Nest, A. Miyake, W. Dür and H.-J. Briegel, quant-ph/0604010.
[4] M. Van den Nest, W. Dür, G. Vidal and H.-J. Briegel, quant-ph/0608060.
[5] R. Raussendorf and H. J. Briegel, Phys. Rev. Lett. **86**, 5188 (2001); Quantum Inf. Comp. **2**(2), 443 (2002).
[6] H. J. Briegel and R. Raussendorf, Phys. Rev. Lett. **86**, 910 (2001).
[7] R Raussendorf, D. E. Browne, and H. J. Briegel, Phys. Rev. A **68**, 022312 (2003).
[8] M. Hein *et al.*, Proceedings of the International School of Physics "Enrico Fermi" on "Quantum Computers, Algorithms and Chaos", Varenna, Italy, July, 2005; quant-ph/0602096.
[9] S.-I. Oum, PhD thesis, Princeton University, 2005.
[10] K. Gödel, Monatshefte für Math. u. Physik **38**, 173-198, 1931.
[11] B. Courcelle and S. I. Oum, Vertex-Minors, Monadic Second-Order Logic, and a Conjecture by Seese, J. Combin, Theory Ser. B, 2006. Accepted.
[12] B. Courcelle, Handbook of graph grammars and computing by graph transformation 1, 313-400. World Sci. Publishing, River Edge, New York, 1997.
[13] Here we adopt a simplified definition of relational structures. We refer to Ref. [12] for the complete definition.
[14] This is proven by using a result in Ref. [9] stating that a graph transformation called *local complementation* can be expressed in C_2MS logic, and by noting that local complementation of graphs corresponds to local unitary operations on the corresponding graph states [8].
[15] To be precise, the definition of decidable C_2MS theory is slightly different from the one adopted here, but the present formulation is known to be equivalent [12].
[16] J.R. Büchi, Proc. Internat. Congr. Logic, Method. and Philos. Sci., 1–12, Stanford, 1960.
[17] M. O. Rabin, Trans. Amer. Math. Soc. 141, 1-35, 1969.
[18] This follows from the fact that the 2D cluster graphs have unbounded rank width [9], and that all families of graphs with unbounded rank width have undecidable C_2MS theories [11].
[19] As $\mathcal{S}_{\text{tri}}$ and $\mathcal{S}_{\text{hex}}$ correspond to universal resources [3], these families have unbounded rank width. Using [11] then again yields the undecidability of the corresponding C_2MS theories.
[20] A conjecture treated in Ref. [11] is particularly relevant in the present context.